What is claimed is:

sub b10 ~~1. In a method for implementing an event transfer system of a real time operating system kernel under a multi-tasking environment in which a priority-based preemptive scheduling is adapted, a method for implementing an event transfer system of a real time operating system kernel which is characterized in that in the case that a plurality of tasks call a kernel system function of receiving an event with respect to one event under the multi-tasking environment, the tasks are blocked and previously inserted into a waiting-list of the event in a higher priority order, and in the case that the event transfer occurs, the task having the highest priority in the waiting-list obtains the event, is woke up and is resumed execution.~~

~~2. The method of claim 1, wherein said waiting-list of the event is managed based on the higher priority order so that the task having the highest priority is arranged at the most leading portion (head) of the waiting-list.~~

~~3. The method of claim 1, wherein when the kernel system function of receiving the event starts, it is checked whether there is an event value already sent.~~

a ~~4. The method of claim 1 ~~or 3~~, wherein as a result of the check whether the event value exists, when the event value exists, the event value is obtained from the event control block buffer, and the task routine is executed by the sort of the event.~~

a

5. The method of claim 1 ~~or 3~~, wherein as a result of the check, when the event value does not exist, the current task is blocked and queued into the waiting-list of the event.

a

2u0b11

6. The method of claim 1 ~~or 3~~, wherein when the kernel system function of receiving the event starts, a step for checking a validity of the event ID is further included for thereby generating an error code in the case of invalidity, and the routine is returned from the kernel system function.

7. The method of claim 5, wherein when the current task is queued into the waiting-list, a time out option is additionally set if it exists.

8. The method of claim 1, wherein when transferring the event, it is checked whether the waiting task exists in the waiting-list of the event.

a

9. The method of claim 1 ~~or 8~~, wherein as a result of the check that whether the task exists in the waiting-list, when the waiting task does not exist, an event value is stored in the event buffer of the event control block.

a

10. The method of claim 1 ~~or 8~~, wherein as a result of the check whether the task exists in the waiting-list, when the waiting task exists, an event value is transferred to the head task of the waiting-list.

2u0b12

11. The method of claim 10, wherein said head task in the waiting-list

which receives the event value is adjusted to the ready state and is inserted into the ready list, and the additional process routine by the sort of the event is executed.

add A1

ad b147